A

PROJECT REPORT

ON

“Automatic Pre-programmed Robotic Arm”

Submitted in partial fulfillment of the award of

DIPLOMA

IN

ELECTRICAL ENGINEERING

Submitted By

Saurabh Sharma	2822064005
Shivam Saxena	1722064017
Rishabh Sharma	2822064004
Dharmendra Kumar Mishra	1722064007
Vishal Saxena	1722064019

SUPERVISIOR	**PROJECT GUIDE**
Er. Mohd Yasir	Er. S.A. Rafey

SESSION 2019-2020

Department of Electrical Engineering

University Polytechnic, Shahjahanpur

INTEGRAL UNIVERSITY, LUCKNOW

CERTIFICATE

This is to certify that this team which is comprised of five students of final year **DIPLOMA IN ELECTRICAL ENGINEERING** in this University Centre is presenting a project on **“Automatic Preprogrammed Robotic Arm”** as our final year project during the session 2019-2020.

This project is completed for the award **of DIPLOMA IN ELECTRICAL ENGINEERING** from Integral University Centre, Shahjahanpur.

(Project Guide)
Er. S.A. Rafey

(Project Supervisor)
Er. MOHAMMAD YASIR

Electrical Engineering Department
Integral University-Lucknow
Campus Shahjahanpur

ACKNOWLEDGEMENT

We would like to express my sincere thanks to my project Guide **Er. S.A. Rafey, Lecturer,** Department of Electrical Engineering, University Polytechnic Shahjahanpur Centre, for his constant support, timely help, guidance, sincere co-operation during the entire period of my work. We are also grateful to him for providing all the necessary facilities during the course of the project work.

We would also like to pay our special thanks to our project supervisor **Er. Mohammad Yasir, Lecturer,** Department of Electrical Engineering, University Polytechnic Campus, Shahjahanpur for his outstanding help provided during various stages of the project.

My special thanks goes to In-charge of our Polytechnic **Dr. M. Naseem &** Head of Department **Er. Mohd. Faraz Khan** for their full support from time to time in making this project. Apart from this **Er. Farazuddin Siddiqui & Mr. Irfan Khan** are also indebted to my heartily thanks.

I am also thankful to my loving parents and my colleagues for their moral support & timely help.

Saurabh Sharma
2822064005

Shivam Saxena
1722064017

Rishabh Sharma
2822064004

Vishal Saxena
1722064019

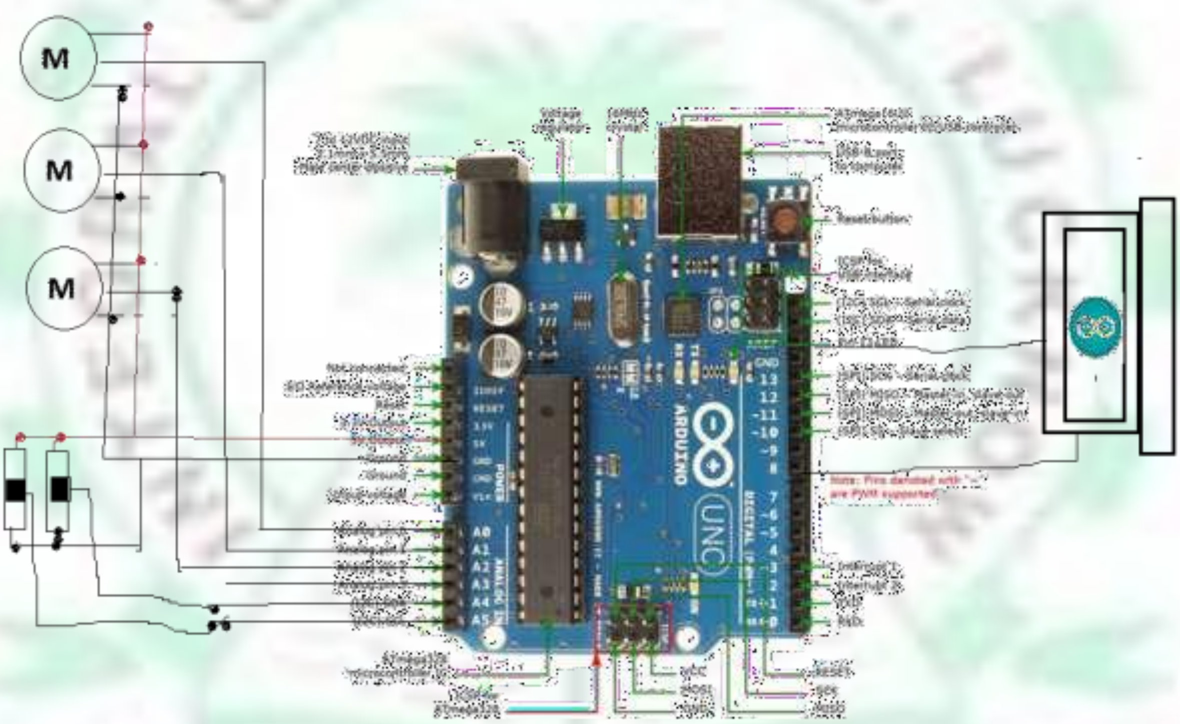
Dharmendra Kumar Mishra
1722064007

ABSTRACT

We present an automated robotic arm capable of performing an accompaniment for a pre-recorded human performance in real time. This project is proposed to the design and development of an automatic robotic arm for use in the assembly of modular buildings elements in the construction site. The automatic system will be developed using this prototype.

This work will based on social robotic arms and computer accompaniment systems using Arduino Uno in order to create more design, interactive, and engaging performances between humans and machines by picking movement or action. We also conduct subjective evaluations on putting or carrying various materials by using robotic arm. Our results show that robotic armsaves our time &manpower. Counter intuitively, such improvement does not exist when the machine is performing a fixed sequence simply follows the machine or copying a human.

CIRCUIT DIAGRAM OF THE PROJECT



COMPONENT LIST

S.N.	NAME OF MATERIALS	SPECIFICATION	QUANTITY
1	MICROCONTROLLER	ATMEGA328	1
2	Arduino Kit	UNO 4.1	1
3	TRANSFORMER	12-0-12	1
4	CAPACITER	470250E	1
		100FU, 25F	2
		PAPER, 104uF	2
5	SERVOMOTOR	FND 7 SEGMENT	3
6	POTENTIOMETER	10K	3
7	LED	-	4
8	RESISTOR	R2, 10k ohm	2
9	SLIDER SWITCH	CH3406, 9600BPS	1
10	TRANSISTER	SMD2N222	1
11	CRYSTAL OSCILLATOR	16MHz	1
12	DIODE	IN4007	1
13	VOLTAGE REGULATOR	LM7805	1
14	Power Supply	**220V**	1
15	RECTIFIER	STD	2
16	Multimeter	STD	1
17	SMPS ADAPTOR	5V, 2A	1
18	INTEGRATED CIRCUIT	IC ESP8266	1
19	USB CABLE	L-COM (2.0)	1
22	SOLDERING IRON	MINIPWRK/TMRK	5
25	GLUE GUN	J2, SMALL	1
26	SOLDERING PASTE	SMALL	1
27	SOLDERING WIRE	SMALL	1

OVERVIEW

Title of the Project:

Automatic Preprogrammed Robotic Arm

Domain:

Power Electronics & Control

PLC & Automation

Robotics

Software:

Arduino 1.8.5

Microcontroller& IC:

89S52
PIC F877A

Power Supply:

230 V AC
12 V DC

Display:

N/A

Transformer:

230 V / 12 V AC

Voltage Regulator

LM 7805

Communication Device:

USB 2.0 Cable (L-COM)

Applications:

For automated Building Construction.
For Self operated cranes.

INTEGRAL UNIVERSITY, LUCKNOW

TABLE OF CONTENTS

CIRTIFICATE **i**

ACKNOWLEDGEMENT **ii**

ABSTRACT **iii**

CIRCUIT DIAGRAM OF THE PROJECT **vi**

COMPONENT LIST **v**

OVERVIEW **vi**

TABLE OF CONTENTS **vii-viii**

Chapter 1: ABOUT PROJECT **1-2**

- 1.1 INTRODUCTION 1
- 1.2 AIM OF THE PROJECT 2
- 1.3 THESIS OF ORGANIZATION 2

Chapter 2: ABOUT TECHNOLOGY **3-8**

- 2.1 INTRODUCTION 3
- 2.2 HISTORY OF ARDUINO 3
- 2.2 DIAGRAM OF ARDUINO 6
- 2.3 FUTURE OF ARDUINO 7

Chapter 3: HARDWARE DESIGN AND WORKING **9-17**

- INTRODUCTION 9
- HARDWARE DESIGING 9
- OPERATING PROCEDURE 16

Chapter 4: SOFTWARE PROGRAMMING OF ARDUINO UNO **18-32**
INTRODUCION 18
PROGRAMMING LANGUGE 19
ARDUINO CODING & IDE 21
MAKE & COMPILE PROGRAM 22

Chapter 5: CONCLUSION & FUTURE SCOPE **33-33**
5.1 CONCLUSION 33
5.2 FUTURE SCOPE 33
REFERENCE **34**

BASE PAPER

INTEGRAL UNIVERSITY, LUCKNOW
L C P E

Chapter 1: ABOUT PROJECT **1-2**

INTRODUCTION 1

AIM OF THE PROJECT 2

THESIS OF ORGANIZATION 2

CHAPTER-1: ABOUT PROJECT

In this chapter the basic introduction about the idea of project including objective & thesis organization is given in short.

INTRODUCTION

Today, technology is developing in the same direction in line with rapidly increasing human needs. The work done to meet these needs makes life easier every day, and these studies are concentrated in robotic arm studies. Robot arms work with an outside user or by performing predetermined commands. Nowadays, the most developed field of robot arms in every field is the industry and medicine sector. Designed and realized in the project, the robot arm has the ability to move around the given directions with 3 servo motors. We can teach any action to this robotic arm, i.e. to pick desired material from one place and put it to another place. While doing this, robot control is provided by connecting to the Arduino application via PC or Mobile with Android App connected to its USB.

This Project represents an Arduino based Pre-programmed robotic arm which is capable to learn any particular given stunt. Now a days, the demand of automatic cranes or Robotic hand is regularly increasing day by day. Then new, advance & efficient method is required to reduce man power. So, we designed this topology of repeating physical action in consecutive manner through Arduino. Through this prototype a fast crane can also be developed to build any civil construction.

Now a days people always needed additional help systems. With the rapid increase in the flow of information, people are now guided to search for different markets and people have entered the competition to manufacture quality products cheaply. Automation systems are also needed to realize this. Because standardized automation systems are required to minimize errors as well as to have experienced and well trained employees for quality products. Because of their physical characteristics, people needed to use auxiliary machines in places where their strength was not enough. These machines, which are operated with the need for human assistance in advance, have been made to operate spontaneously without the need of human power with the progress of technology. One of the most used components of automation systems is robots. Robotic systems; Mechatronics Engineering, Mechanical Engineering, Electrical Engineering and Computer Engineering have all come together to work together. In the project, we have been done and implemented in order to have knowledge about mechanics and software during the operations carried out by the robot arm which is designed to fulfill the tasks determined in accordance with predetermined commands.

First, it was determined what function the robot arm would be and what movements it could make. Robotic arm made of Android phone or tablet control; it can carry the desired material, mix it up and perform the commands previously determined by a user. If this project is also a designated task; the robotic arm takes a piece of material and brings it to the desired position and then records its movements and lets it do the same action until we stop it. The servomotor is preferred in order to be able to perform these operations properly since the motor to be selected must operate precisely and must be at morderate torque.

AIM OF THE PROJECT

The main aim of this project is save man power to achieve modern control of Cranes or Robotic Hands by Using Arduino programming. This is mainly done by the controlling the servomotor thorough Arduino Commands. By analyzing the Arduino program, our system controls these servomotors as per the given action or Pre-Programmed Command.

In earlier years we use manually robotic arms for various purpose but this technology of Teach and Play is new in India. Robotic arms, many areas are developable. Thanks to the robotic arms, many tasks are made easier and the resulting error level has been reduced to a minimum. For example; some pharmacy-based drug-giving robots and a projected robot arm have been developed.

it can be used in a wide range of applications from the Civil sector to the automation systems. With the robotic arms developed in this way, the risk of 1 Axis + 1 Axis - 2 Axis + 2 Axis - 3 Axis + 3 Axis - 4 Axis + 4 Axis and so on. Despite the fact that the robotic arm made by this project is of prototype quality, it has a quality that can be improved for more robotic systems. Besides these, robotic arm sector, which is open to development, will keep its importance in the future.

THESIS OF ORGANIZATION

The structure of the entire thesis organization is as following:

Chapter -1: the basic introduction about the project including the requirement of tap changing in transformer and thesis organization is given in short.

Chapter -2: In this chapter the basic introduction about the design & working of project including its all important hardwares such as part mounting, making an arm & SOPs etc.

Chapter -3: In this chapter the AURDINO Technology is discussed in detail alongwith its basics & future scope in india.

Chapter -4 In this chapter the software modulation, programming language, coding etc are described for Arduino UNO 1.8.2 through sketching out its programs & blocks.

Chapter -5: This chapter is the valedictory of our project thesis in which we wrote the final conclusion of our Project illustrating the result along-with the short discussion about the future scope of our presented model.

We also included a separate section for Reference & Appendix in this Thesis.

Chapter 2: ABOUT TECHNOLOGY 3-8

INTRODUCTION 3

HISTORY OF ARDUINO 3

DIAGRAM OF ARDUINO 6

FUTURE OF ARDUINO 7

CHAPTER-2: ABOUT TECHNOLOGY

In this chapter the basic introduction about the idea of project including objective & thesis organization is given in short.

INTRODUCTION

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. This is designed to make electronics more accessible to artists, designers, hobbyists and anyone interested in creating interactive objects or environments.

In short the following points can better describe Arduino technology:

- ✓ The Arduino is an open-source electronics platform based on easy-to-use hardware and software used to build electronics projects.
- ✓ All Arduino boards have one thing in common which is a microcontroller. A microcontroller is basically a really small computer.
- ✓ With the Arduino, you can design and build devices that can interact with its surroundings. The Arduino boards are basically a tool for controlling electronics. They are able to read inputs with their onboard microcontroller (eg. Light on a sensor, an object near a sensor) and turn it into an output (Drive a motor, ring an alarm, turning on an LED, display information on an LCD).
- ✓ However, to do this, you will first have to program the Arduino board.
- ✓ With the Arduino, makers and electricians can easily prototype their products and make their ideas come to life.

HISTORY OF ARDUINO

Arduino was born at Ivrea Interaction Design Institute (IIDI) as an easy tool for fast electronic prototyping, aimed at students without a background in electronics and programming. However, the demand for Arduino design spec started to change after reaching a wider community. For example, changed in 8-bit microcontroller with the 32-bit microcontroller. Today, there are more than 20 different types of official Arduino in the market. (Exclude the mod)

In 2005, building upon the work of Hernando Barragán (creator of Wiring), Massimo Banzi and David Cuartielles created Arduino, an easy-to-use programmable device for interactive art design projects, at the Interaction Design Institute Ivrea in Ivrea, Italy. David Mellis developed the Arduino software, which was based on Wiring. Before long, Gianluca Martino and Tom Igoe joined the project, and the five are known as the original founders of Arduino. They wanted a device that was simple, easy to connect to various things (such as relays, motors, and sensors), and easy to program. It also needed to be inexpensive, as students and artists aren't known for having lots of spare cash. They selected the AVR family of 8-bit microcontroller (MCU or µC) devices from Atmel and designed a self-contained circuit board with easy-to-use connections, wrote bootloader firmware for the microcontroller, and packaged it all into a simple integrated development environment (IDE) that used programs called –sketches.‖ The result was the Arduino. Since then the Arduino has grown in several different directions, with some versions getting smaller than the original, and some getting larger. Each has a specific intended niche to fill. The common element among all of them is the Arduino runtime AVR-GCC library that is supplied with the Arduino development environment, and the on-board bootloader firmware that comes preloaded on the microcontroller of every Arduino board.

The Arduino family of boards use processors developed by the Atmel Corporation of San Jose, California. Most of the Arduino designs utilize the 8-bit AVR series of microcontrollers, with the Due being the primary exception with its ARM Cortex-M3 32-bit processor. We don't cover the Due in this book, since it is radically different from the AVR devices in many fundamental ways and really deserves a separate discussion devoted to it and similar microcontrollers based on the ARM Cortex-M3 design. Over the years the designers at Arduino.cc have developed a number of board designs. The first widely distributed Arduino board, the Diecimila, was released in 2007, and since its initial release the Arduino family has evolved to take advantage of the various types of Atmel AVR MCU devices. The Due, released in 2012, is the first Arduino to utilize a 32-bit ARM Cortex-M3 processor, and it breaks from the rest of the family in terms of both processing power and board pinout configuration. Other boards, like the LilyPad and the Nano, also do not have the same pinout as the other members of the family, and are intended for a different range of applications—wearables in the case of the LilyPad; handheld devices for the Esplora; and compact size in the case of the Mini, Micro, and Nano. The following table can easily describe the year wise history of Arduino:

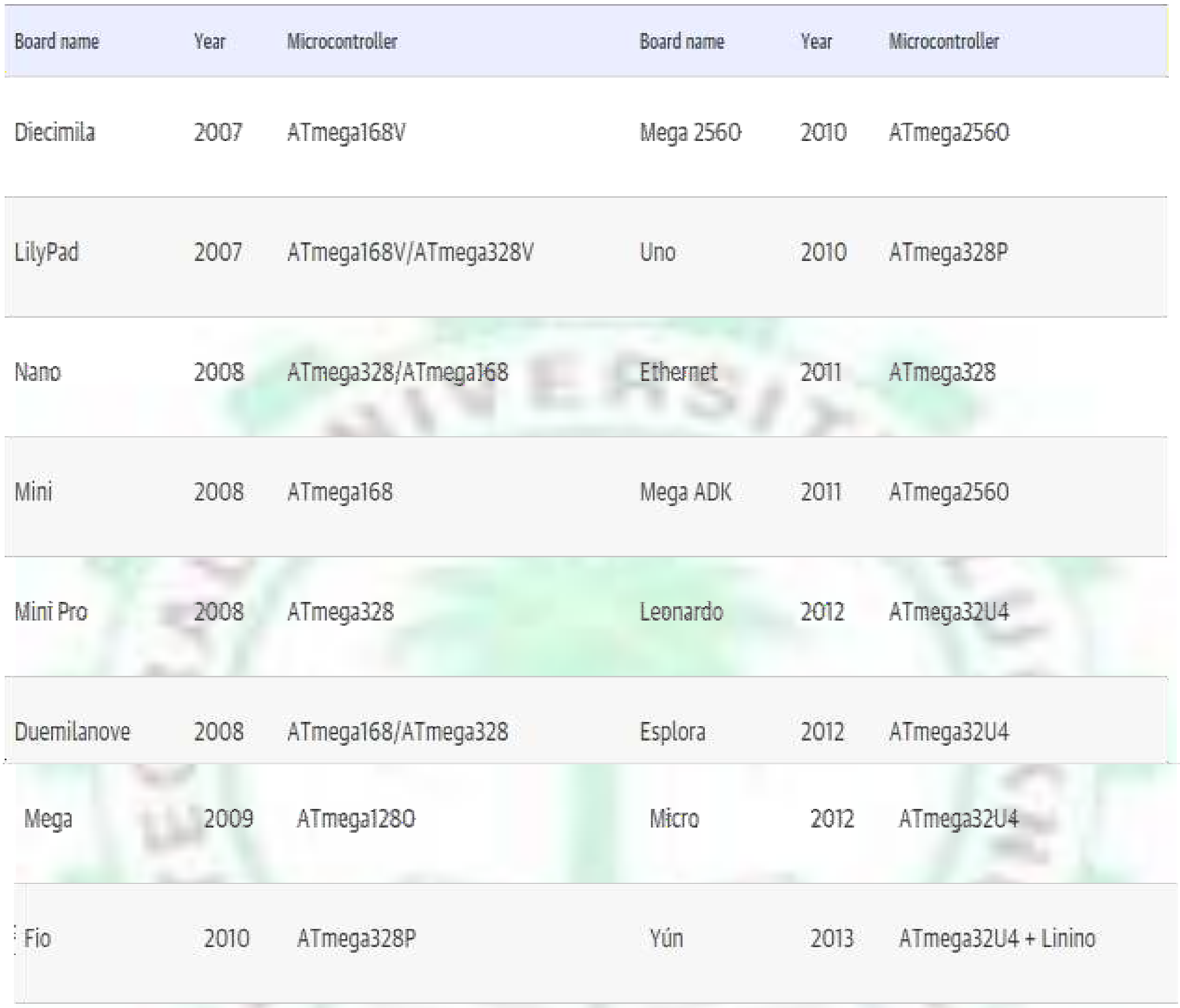


Table Timeline of Arduino products

Board name	Year	Microcontroller	Board name	Year	Microcontroller
Diecimila	2007	ATmega168V	Mega 2560	2010	ATmega2560
LilyPad	2007	ATmega168V/ATmega328V	Uno	2010	ATmega328P
Nano	2008	ATmega328/ATmega168	Ethernet	2011	ATmega328
Mini	2008	ATmega168	Mega ADK	2011	ATmega2560
Mini Pro	2008	ATmega328	Leonardo	2012	ATmega32U4
Duemilanove	2008	ATmega168/ATmega328	Esplora	2012	ATmega32U4
Mega	2009	ATmega1280	Micro	2012	ATmega32U4
Fio	2010	ATmega328P	Yún	2013	ATmega32U4 + Linino

Above table is not a buyer’s guide. It is provided to give a sense of historical context to the Arduino. As you can see, the years 2007 and 2008 saw the introduction of the LilyPad; the small form-factor boards like the Nano, Mini, and Mini Pro; and the introduction of the Duemilanove as a natural evolutionary step based on the Diecimila. While there are no significant physical differences between the Diecimila and the Duemilanove, the Duemilanove incorporates some refinements in the power supply, most notably in its automatic switchover between USB power and an external DC (direct current) power supply. Later versions of the Duemilanove also utilize the ATmega328 MCU, which provides more memory for programs.

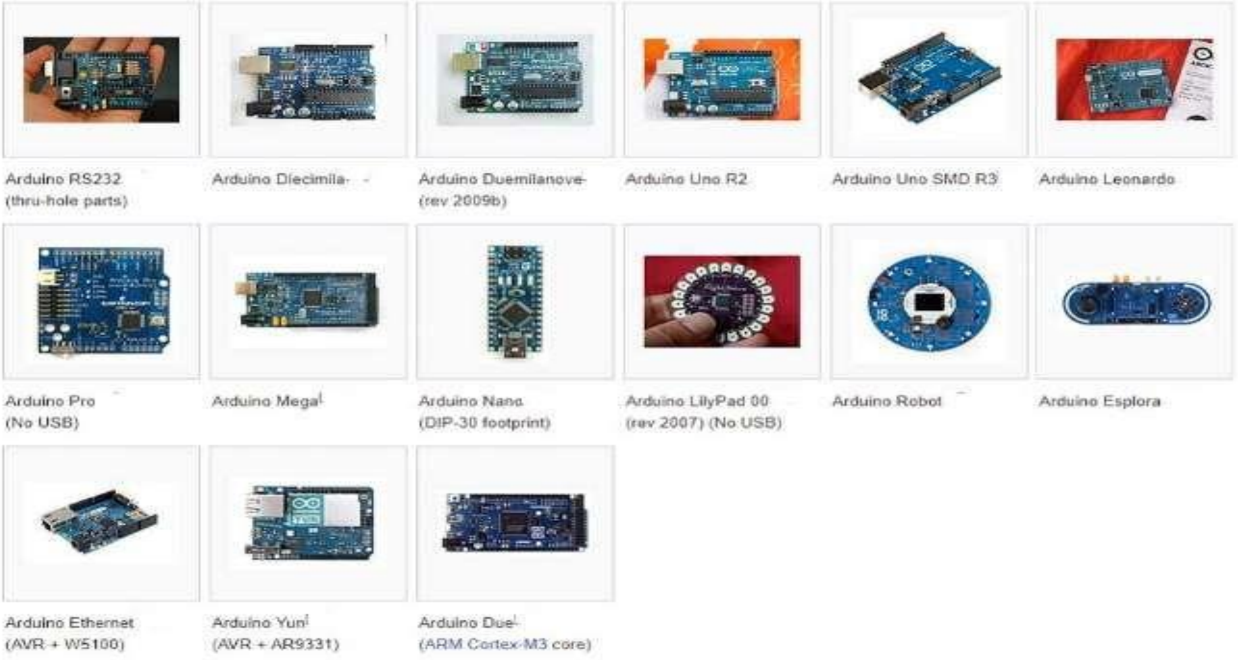


Figure (2.1):Different types of Ardino Available in Market

DIAGRAM OF ARDUINO

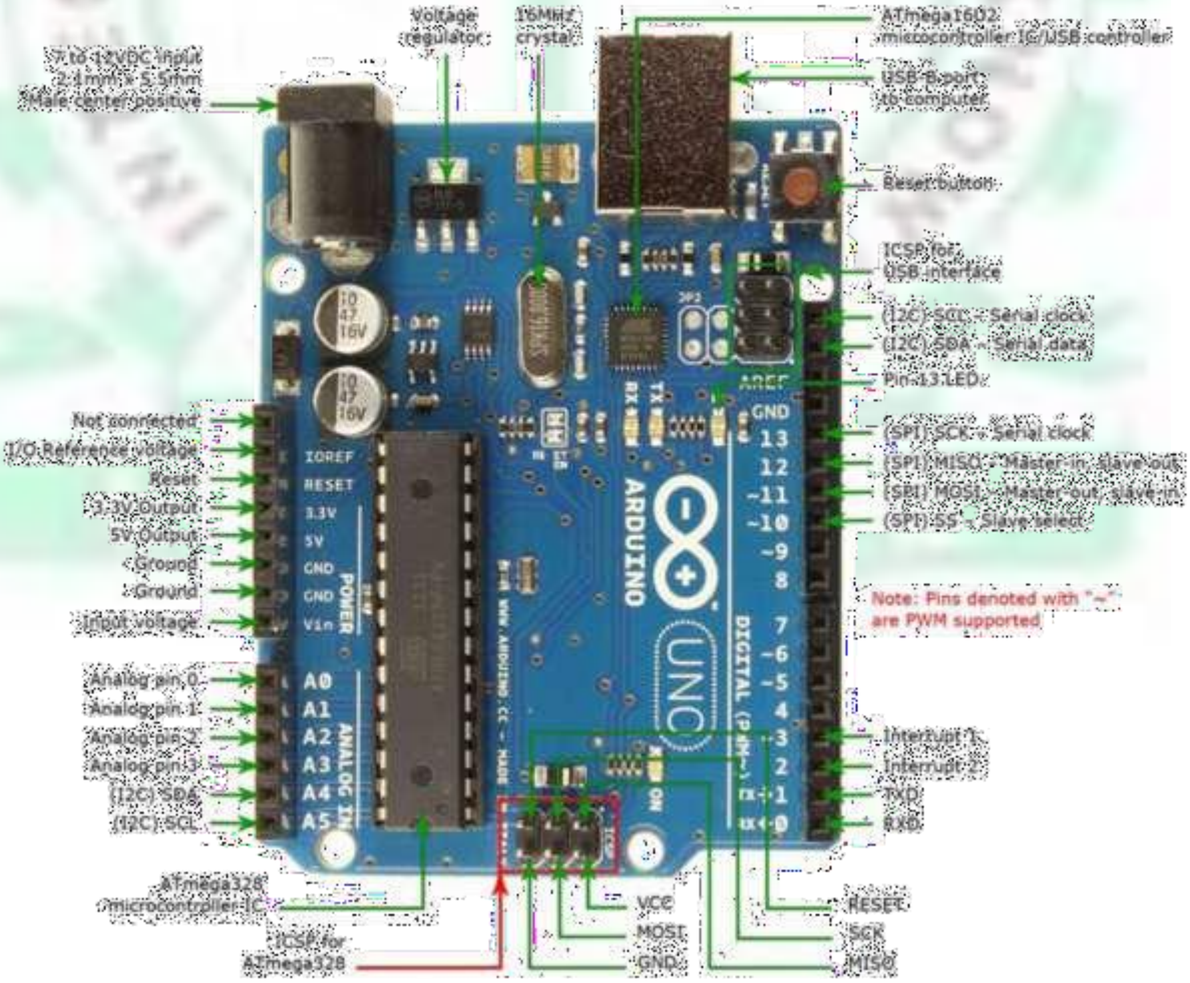


Figure (2.2): Arduino UNO SMD Pin diagram

In the above figure each and every pin of Arduino kit is described through these ports we can easily connect as per requirement. The UNO is the best board to get started with electronics and coding. If this is your first experience tinkering with the platform, the UNO is the most robust board you can start playing with. The UNO is the most used and documented board of the whole Arduino & Genuino family. So a broader overview is presented here as follows:

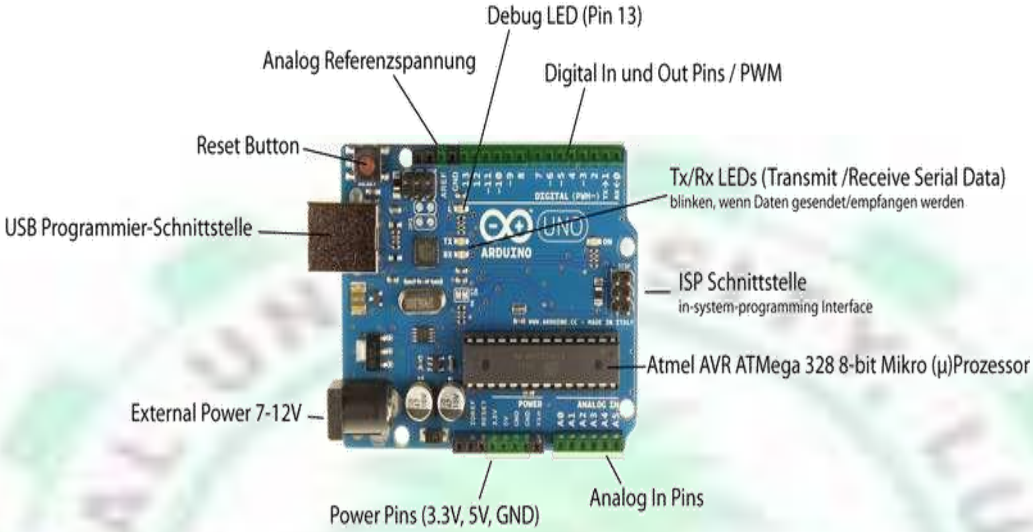

Debug LED (Pin 13)
Analog Referenzspannung
Digital In und Out Pins / PWM
Reset Button
Tx/Rx LEDs (Transmit /Receive Serial Data)
blinken, wenn Daten gesendet/empfangen werden
USB Programmier-Schnittstelle
ISP Schnittstelle
in-system-programming Interface
Atmel AVR ATMega 328 8-bit Mikro (µ)Prozessor
External Power 7-12V
Analog In Pins
Power Pins (3.3V, 5V, GND)

Figure (2.3): Arduino UNO SMD Overview

2.3 FUTURE OF ARDUINO

At this point in time many businesses haven't heard of and aren't using the Arduino, but I believe that will change pretty quickly as businesses eventually hire people familiar with Arduino. One of the major things Arduino is going to be able to do for business is to reduce the cost of prototyping, allowing companies to iterate more during development, leading to better, more functional products. The Arduino is going to enable businesses to do things that aren't commonly done today with remote sensor networks. This could lead to entirely new control strategies for making buildings more comfortable, saving energy, and reducing maintenance costs for equipment. The Arduino is going to allow businesses to develop products that are more easily upgradeable. Right now if you buy a product, such as a microwave, there's no way to change the functionality. If the microwave used an Arduino board, you would be able to change the interface or the way that the microwave cooked food to suit your desires. The Arduino is going to reduce the minimum volume necessary to include a control and sensing system with a product. Instead of spending large amounts of money to build hundreds of inflexible circuit boards, the Arduino will allow businesses to bring many more unique devices to market at lower breakeven volumes. We'll see a lot more lower-volume customized products. The Arduino is going to allow developing countries to do things that they couldn't in the past. From medical devices to low-cost PLC controllers, the Arduino is going to open up a whole host of options and capabilities for medicine and manufacturing in areas where regulation isn't as stringent. Proprietary electronics like the boards that control your washing machine could certainly be disrupted by the Arduino. Instead of being held hostage by a company that wants to charge a lot of money for a replacement, you might alternatively be able to replace a failed component with a custom Arduino board. Or you might be able to use an Arduino board to diagnose exactly what's wrong with your washing machine much more easily than otherwise. Maybe the company building the washing machine will sell you a "glider" and the open-source community will develop a control system for it. So we'll likely see control systems for consumer appliances that are much more customized and user-friendly developed by the open-source community and networkable. Things will be much easier to use and do what we want without us having to figure out which 20 buttons to push. And you'll be able to monitor and control your appliances remotely.

Nowadays, technology has dominated, to a great degree, our lives. One of its ―miracles‖ is the Arduino. Most people probably don't know what Arduino technology is or, even if they do, they have not fully understood its full potential. However, its range of operations makes it a useful tool for the future. As Edward Teller once stated: ―the science of today is the technology of tomorrow.‖ Hence, the capabilities and the way this small scientific ―miracle‖ works need to be described and communicated sufficiently.

Arduino is an open-source computer hardware and software company, a project and a user community that designs and manufactures kits with a scope of creating digital devices and interactive objects that can sense and control the physical world [1]. Arduino programming language is based on Wiring, which contains some data from C and C++ as well. In other words, it is a platform with which the correct programming can contribute in the development of robotics. In order to program Arduino, computer software provided by the company should be downloaded and then users can start creating projects with this platform. The simplest thing you can do is to make an LED blink automatically via a computer [2]. For more complex applications like aiding vision for patients, you also need various ―boards‖ such as Arduino Uno [3], Robot [4], Pro [5], and Mega 2560 [6]. In general, whoever has imagination and patience, can manufacture whatever he/she wants using Arduino. Moreover, no advanced programming knowledge and techniques are needed – basic knowledge is adequate.

Many medical problems people face today are difficult to solve, but Arduino may provide helpful solutions. Arduino is capable of making medical devices and low-cost PLC controllers. Such devices will open up a whole host of options and capabilities that never existed for medicine. Specifically, these methods may expedite simple examinations such as blood pressure, so that doctors can deal with more complex problems such as treating cancer or attending to patients in need of emergency care. In addition, Arduino can be used to develop an infrared heart sensor. This device would be used on a subject's finger to measure blood flow through the patient's finger. The amount of oxygenated blood will be shown in the finger, causing the infrared light to reflect off the skin and towards the transmitter. A sensor wired to an Arduino will show the fluctuations of the oxygen and detect the heartbeat of the subject. Lastly, Arduino controlled devices can assist the elderly by prompting them to take their pills.

In business, Arduino can allow businesses develop more products that are easily upgradeable. For example, there is no way for consumers to alter the functionality of a washing machine or a microwave they purchased. Therefore, if such products used an Arduino board, then consumers would be able to modify the interface in a way to meet their needs. Furthermore, this board could reduce the minimum volume necessary to include a control and sensing system within consumer goods and allow businesses to bring many more unique devices to the market at lower volumes. Whether it's medicine, business, or industry, the endless possibilities enabled by Arduino makes it the technology of the future.

In short, the future for Arduino is very bright, and it's probably about where programming and the Internet was about 20 years ago. We've had a lot of people who were just hobbyists dabbling with programming and the Internet, and now we have huge open-source collaborative projects and game-changing functionality that was unavailable 20 years ago.

Chapter 3: HARDWARE DESIGN AND WORKING 9-17

INTRODUCTION 9

HARDWARE DESIGING 9

OPERATING PROCEDURE 16

CHAPTER-3: HARDWARE DESIGN AND WORKING

In this chapter the basic introduction about the design & working of project including its all important hardware such as part mounting, making an arm & SOPs etc.

INTRODUCTION

Hardware designing & working procedure is an integral part of each and every project. Though the ultimate success of our project also depends upon both on its software and hardware but the hardware design is usually an initiative before software programming. Computer scientists pride themselves in their ability to develop quality software. Similarly electrical engineers are well-trained in the processes to design both digital and analog electronics. Manufacturers, in an attempt to get designers to use their products, provide application notes for their hardware devices. The main objective of this class is to combine effective design processes together with practical hardware techniques in order to develop our prototype. These hardware skills include modular design, layered architecture, abstraction and verification by using good mechanical workshop skills. In general our system must be small end of the size scale, but never the less these systems can be quite complex. Though both hardware and software skills are essential for developing this prototype but hardware technique is the basic backbone. Designing good model is an art that must be developed through regular workshop practice and cannot be added on at the end of a project. Just like any other discipline (e.g., music, art, science, religion), expertise comes from a combination of study and practice. The watchful eye of a good mentor can be invaluable, so we take the risk and show our skill to others inviting praise and criticism. So in this chapter, we will begin by presenting a general approach of electronic hardware designing along with its successful demonstration or working.

HARDWARE DESIGNING

Various Materials are needed for making mechanical part of this robot arm which were supplied, and then the materials like pastic box, scale, glue gun connecting wires etc were drawn on Solid Works of Basic structure and all the mechanical part was assembled with these materials. In short there are following major credit in which the hardware designing can be explained:

a) Part Mounting
b) Making Robotic Arm
c) Servomotor Assembly
d) Potentiometer Assembly
e) SMPS Assembly

a) *Part Mounting:*

The cut parts are assembled together with the servo motors and assembly of the robot arm is completed. The robot arm moves by 4 axes and performs this movement with 5 Mini Servo Motors. After drawing the parts on Solid Works and making the necessary parts, the design of the mechanical part is started and the robot arm is made of a stationary lower body and movable upper

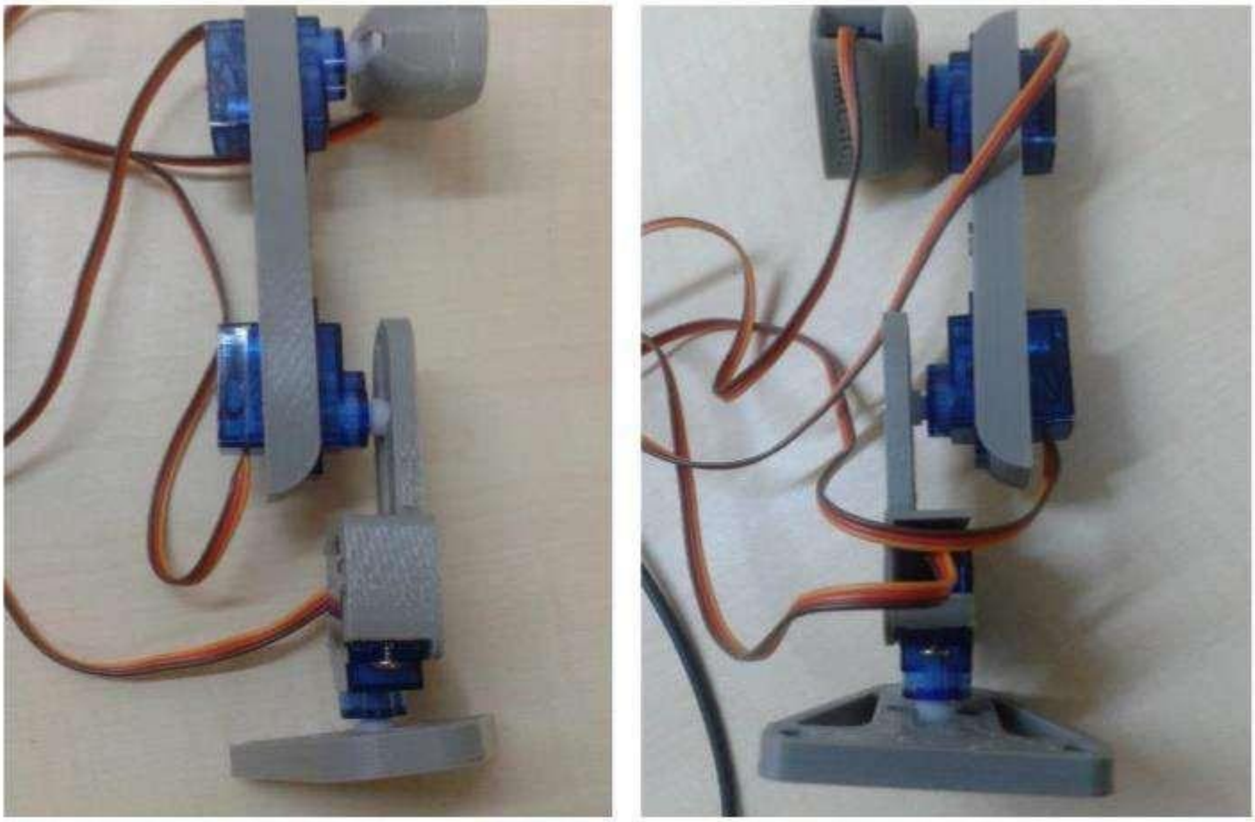


Figure (3.1): Mounting of servo motors in 1st, 2nd, and 3rd axes

body. There is one servo motor in the stationary lower part of the robot arm and this robot is called axis 1. The 1st axis provides rotation of the robot arm to the right or left. There are 2 servo motors in the moving upper body part. Since these servo motors must operate parallel to each other, both start and end positions are set simultaneously. These two servo motors are named as 2nd axis and 3rd axis respectively. The tasks of 2nd and 3rd axes are to move the robot arm up and down. Mounting of servo motors in 1st and 2nd and 3rd axes is given in figure.

Figure (3.1): Base used for Part Mounting

Steps for Part Mounting are as follows:

Step 1: Plastic Box is used as a Base Material

Step 2: Cutting & Jointing Arms by using Glue gun & Scale

Step 3: Assembling of Servomotor

Step 4: Apply Pinch Mechanism

Step 5: Making Base to install Robotic Arm

Step 6: Circuit Connection by using wires & install SMPS

Step 7: Apply Arduino Kit & install USB

Step 8: Demonstrate through Teach & Play.

Figure (3.2): Glue Gun used for Part Mounting

b) *Making an Arm:*

Kinematics in robotics is the science of motion investigation. Robot arm links can be rotated or offset according to the reference coordinate frame. A systematic and general approach developed by Denavit and Hartenberg establishes the relationship between the robot endpoint and the total displacement of robot arm links. Angular and linear displacements between limbs are called joint coordinates and are defined by limb variables. In order to determine the amount of rotation and displacement according to the reference coordinate system of the endpoint, the matrices A which represent the amounts of each limb rotation and displacement are multiplied in turn. If the coordinates of the end point are given, limb variables can be obtained by going backward. These operations are called forward and inverse kinematics. The next section will explain how to determine forward and reverse kinematics. The general transformation matrix can be quite complex even for simple robots. It can be found in standard textbooks such as the Jacobian matrix for

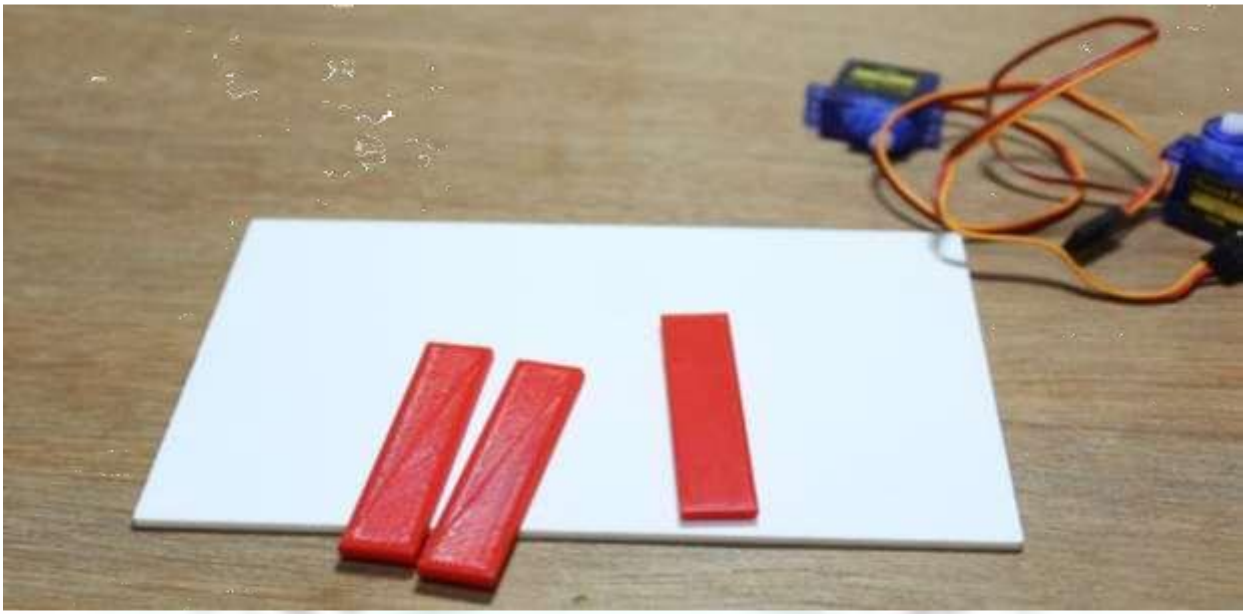


Figure (3.3): Material Used for Robotic Arm Mounting

standard robotic arms. All the internal programming mechanism is govern through the matrixes a sits internal calculation of program.

c) *Servomotor Assembly:*

Servo motor as J1, J2, J3, J4 fix the servo motors as shown in image use 3M tape to glue servo, use thin flexible plastic strip to make griper, make hole in center of each finger tie thread in that hole pass this thread from center hole and tie knot at the other end of thread with 4th servo motor's knob, as you stretch thread finger get close vise versa. Fix whole arrangement on strong rigid base. Servo; Detects the operation error of a mechanism, provides feedback and corrects faults. The servo motor can have alternating current (AC), direct current (DC) or stepper motors. In addition to these, there are drive and control circuits. Servo motors are the kinds of motors that can fulfill the commands we want. They can operate steadily even at very small or very large speeds. In these motors, the large moment can be obtained from the small size.

Servo motors are used in control systems such as fast operation, excessive axis movement, condition control and so on. Servo motors are the last control element of a mechanism. They are highly sensitive and servo motors are used in conjunction with electronic or programmable circuits. These engines are divided into AC and DC. When the AC servo motors are brushless type motors, the servo motors brush. Servo motors are mostly three cables. These are a red cable for power, black for grounding and yellow cables for control (data, data). One of the servomotors used in the production phase of the project is shown in Fig.

Figure (3.4): Servomotor Assembly

In the project Tower, Pro SG90 Mini servo motor is used. Some features of this servo motor; versatile operation, 10 µs pulse width control, V_{P-P}: 3-5 V

Square wave and working voltage of 4.8-6V. The used servomotor has a working voltage of 0.12 s / 60 ° and a torque of 1.2-1.6 kg/cm at low operating voltage.

Servo motors are controlled according to the signal condition. In doing so, the supplied pulse width modulated (PWM) signal is used with the data bus. Each servo motor is controlled by a PWM signal at 10-20 ms and at 0.5-1.5 ms.the position of the motor shaft is determined according to the duration (tk) of this signal at logic 1. These;

- When $t_k = 0.5$ ms, the motor shaft rotates to the end,
- When t_k= 0,5- 1 ms, the position of the motor shaft is in the middle,
- When t_k= 1 - 1.5 ms, the motor shaft turns to the right,
- When t_k= 10-20 ms (when the same signal is given again) it remains in its old position,

The position control of these motors is determined using the required pulses. The servo motors DC used in the project are kept at about 5V during operation.

d) Potentiometer Assembly:

Fix potentiometer as shown in figure name potentiometer as do previous R3, R4, R5, R6 this time place R6 separately for easy access this potentiometer control gripper to pick and place. Potentiometer arrangement symmetry must be same as servo arm. Fix whole arrangement on strong rigid base.

Figure (3.5): Potentiometer

e) SMPS Assembly:

A switched-mode power supply (SMPS) is an electronic circuit that converts power using switching devices that are turned on and off at high frequencies, and storage components such as inductors or capacitors to supply power when the switching device is in its non-conduction state.

Basically SMPS essentially has

1. Input rectifier
2. Inverter
3. Voltage converter
4. Output regulator

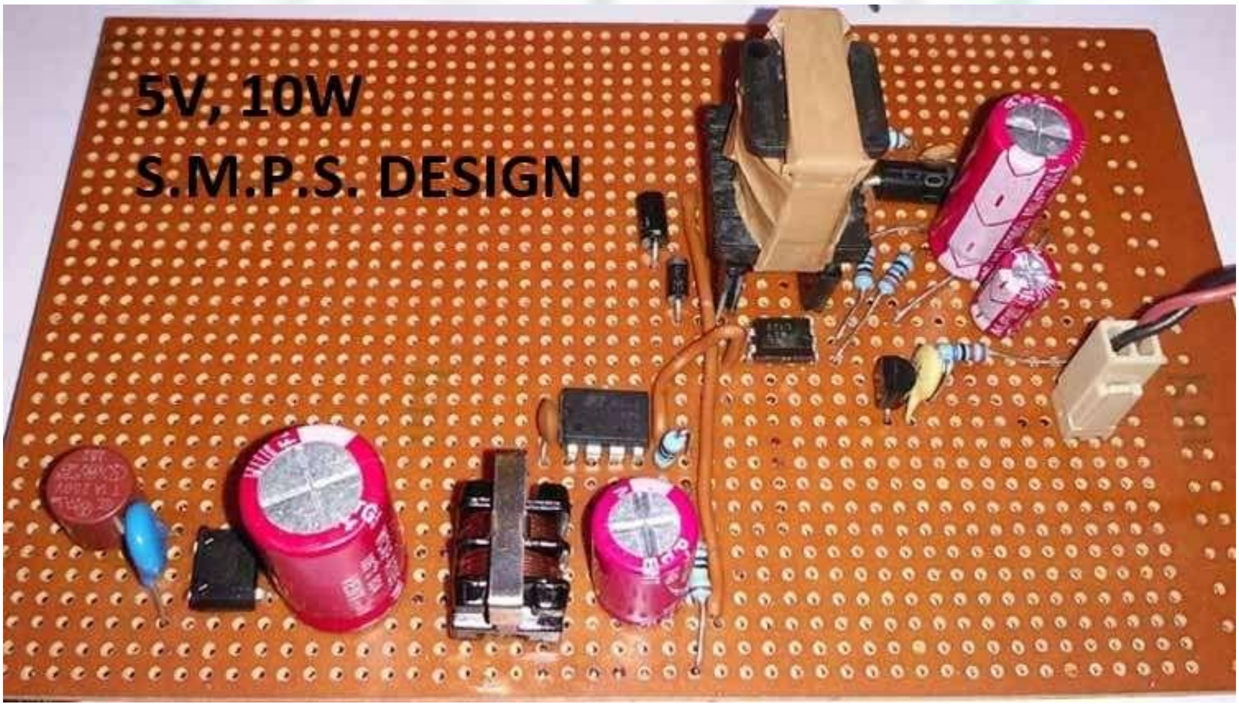
In short we can say that SMPS uses a **switching regulator** to convert electric power efficiently. SMPS transfers electric power from a source (AC mains) to the load by converting the characteristics of current and voltage. SMPS always provide a well regulated power to the load irrespective of the input variations. SMPS incorporates a **Pass transistor** that switches very fast typically at **50Hz** and **1 MHz** between the on and off states to minimize the energy waste. SMPS regulates the output power by varying the on to off time using minimum voltage so that efficiency is very high compared to the linear power supply.

In our Project Servo inputs, Arduino pin inputs, and USB Cable inputs. Servo motors are activated in this way, the USB of Arduino, Nano connections and power supply connections are connected. Thanks to this SMPS circuit assembly we use, it is possible to distribute the 5V from the power supply to the servo motors.

The power supply selected for feeding the control circuit of the servomotors is capable of delivering the same current even if all the synchronous servomotors are operating. When all servo motors are operated at the same time, they draw 2A current. In addition, 5 V was needed for the Arduino used for robot movement in the project. This requirement is provided by a 5V power supply.

Figure: 5V 2A SMPS Power Supply Circuit

Power Supply Unit (PSU) is a vital part in any electronic product design. Most household electronic products like Mobile Chargers, Bluetooth Speakers, Power Banks, Smart Watches etc requires a Power Supply circuit that could convert the AC mains supply to 5V DC to operate



them. In this project we will build a similar **AC to DC power supply circuit** with 10W power rating. That is our circuit will convert the 220V AC mains to 5V and provide a maximum output current upto 2A. This power rating should be enough to power most electronic products running on 5V. Also **5V 2A SMPS circuit** is quite popular in electronics since there are lots of microcontrollers which operates on 5V.

Step by step method to design 5V DC power supply

Step 1: The selection of regulator IC. ...
Step 2: The selection of transformer. ...
Step 3: The selection of diodes for bridge. ...
Step 4: The Selection of smoothing capacitor and calculations. ...
Step 5: Making the power supply safe.

The idea of the project is to keep the build as simple as possible, hence we will design the complete circuit over a dotted board (perf board) and will also build our own transformer so that anyone could replicate this design or build similar ones. Excited Right! So lets get started. Previously we have also built a 12V 15W SMPS circuit using a PCB, so people who are interested in how to design a PCB for a PSU project (power supply unit) can check that as well.

Different varieties of power supply behave differently in different environments. Also, SMPS works in specific input-output boundaries. Proper specification analysis needs to be performed before going forward with the actual design. But for this project the following Design Specifications are used:

a) Input specification:

This will be an SMPS in AC to DC conversion domain. Therefore, the input will be AC. For the input voltage value, it is good to use a universal input rating for the SMPS. Thus, the AC voltage will be 85-265VAC with 50Hz rating. This way the SMPS can be used in any country irrespective of their AC mains voltage value.

b) Output specification:

The output voltage is selected as 5V with 2A of the current rating. Thus, it will be **10W output**. Since this SMPS will provide **constant voltage** irrespective of the load current, it will work on the CV (Constant Voltage) mode. This output voltage of 5V should be constant and steady even at the lowest input voltage during a maximum load (2A) across the output.

It is highly desired that a good power supply unit has a **ripple voltage of less than 30mV pk-pk**. The targeted ripple voltage for this SMPS is less than 30mV peak-peak ripple. Since this SMPS will be built in veroboard using a **handmade switching transformer** we can expect a slightly higher values of ripple. This problem can be avoided by using a PCB.

c) Protection features:

There are various protection circuits that can be employed in a SMPS for a safe and reliable operation. The protection circuit protects the SMPS as well as the associated load. Depending on the type, protection circuit can be connected across input or across the output.

For this SMPS, **input surge protection** will be used with a maximum operating input Voltage of 275VAC. Also, to deal with EMI issues, a **common mode filter** will be used for blanking out the generated EMI. On the Output side we will include **short circuit protection**, **over-voltage protection**, and **over-current protection**.

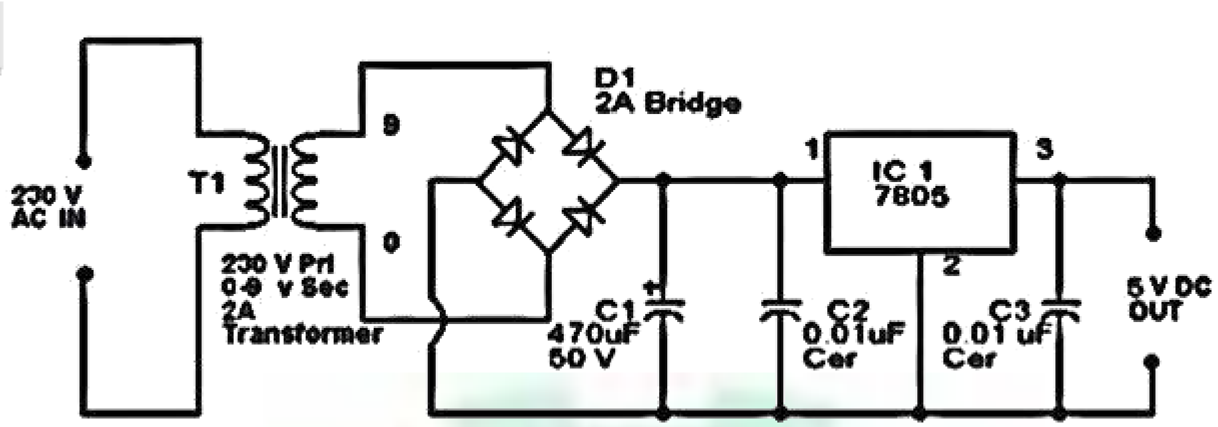


Figure (3.6): SMPS Adapter Assembly (Circuit Diagram)

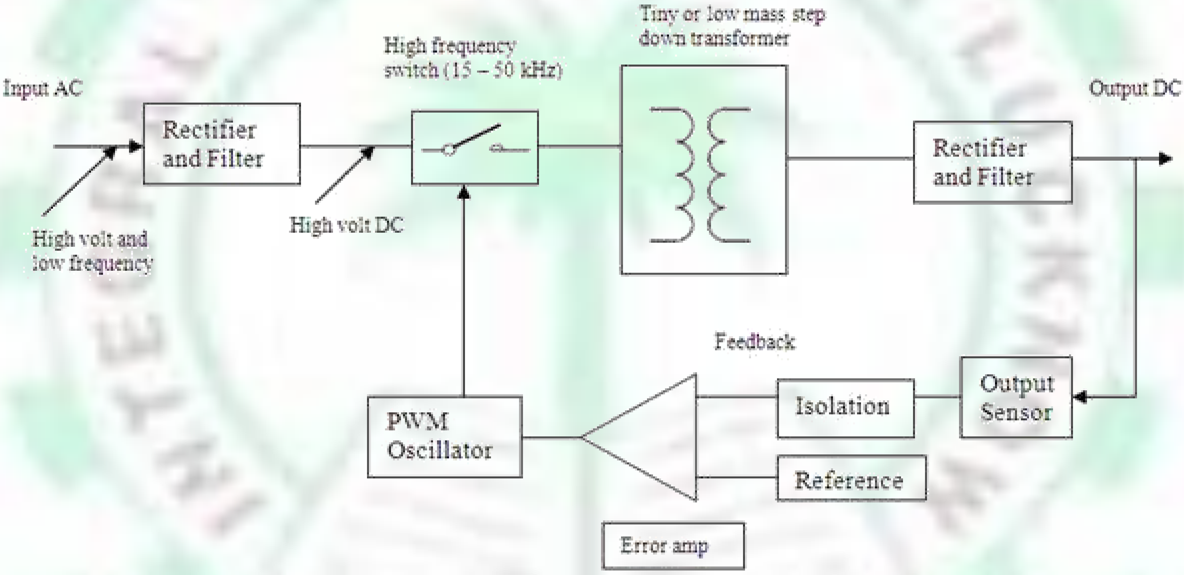


Figure (3.7): SMPS Adapter Assembly (Block Diagram)

OPERATING PROCEDURE

In operating process we are describing instructable steps for demonstrating this prototype. This will show how to operate this pre-programmable robotic arm with record and play function through Arduino software. This movement action works on three mini-servo with the help of three potentiometer which will read from potentiometer is record to Arduino board in teach mode and those steps are played in loop & in play mode. To make this project run smoothly potentiometers must be varied in small steps.

A standard operating procedure (SOP) is a set of step-by-step instructions compiled by an us to help for carrying its complex working. SOPs aim to achieve efficiency, quality output and uniformity of performance, while reducing miscommunication and failure to comply with industry regulations

For this standard operating procedure overall sequential steps can be summarized as follows:

Step 1: Connect USB to PC or Mobile in which Arduino software is pre-installed

Step 2: Connect the power supply by using SMPS adapter

Step 3: Open Serial Monitor in Arduino Software & Put Record command i.e. ‘R’

Step 4: Record your action by using three potentiometer control knobs

Step 5: Put Play command i.e. ‘P’ & observe pre-recorded action.

Step 6: Press reset button to stop this Robotic

During step 3, step 4 & Step 5 various matrixes are observed on Serial Monitor screen of Arduino software, this is nothing but the internal calculations of its program to rotate its servomotors in order to follow its degree of freedom around particular axis. As it is concerned that there are three degree of freedoms in order to play its teaching actions.

INTEGRAL UNIVERSITY, LUCKNOW
L C P E

Chapter 4: SOFTWARE PROGGRAMMING OF ARDUINO UNO 18-32

INTRODUCION 18

PROGRAMMING LANGUGE 19

ARDUINO CODING & IDE 21

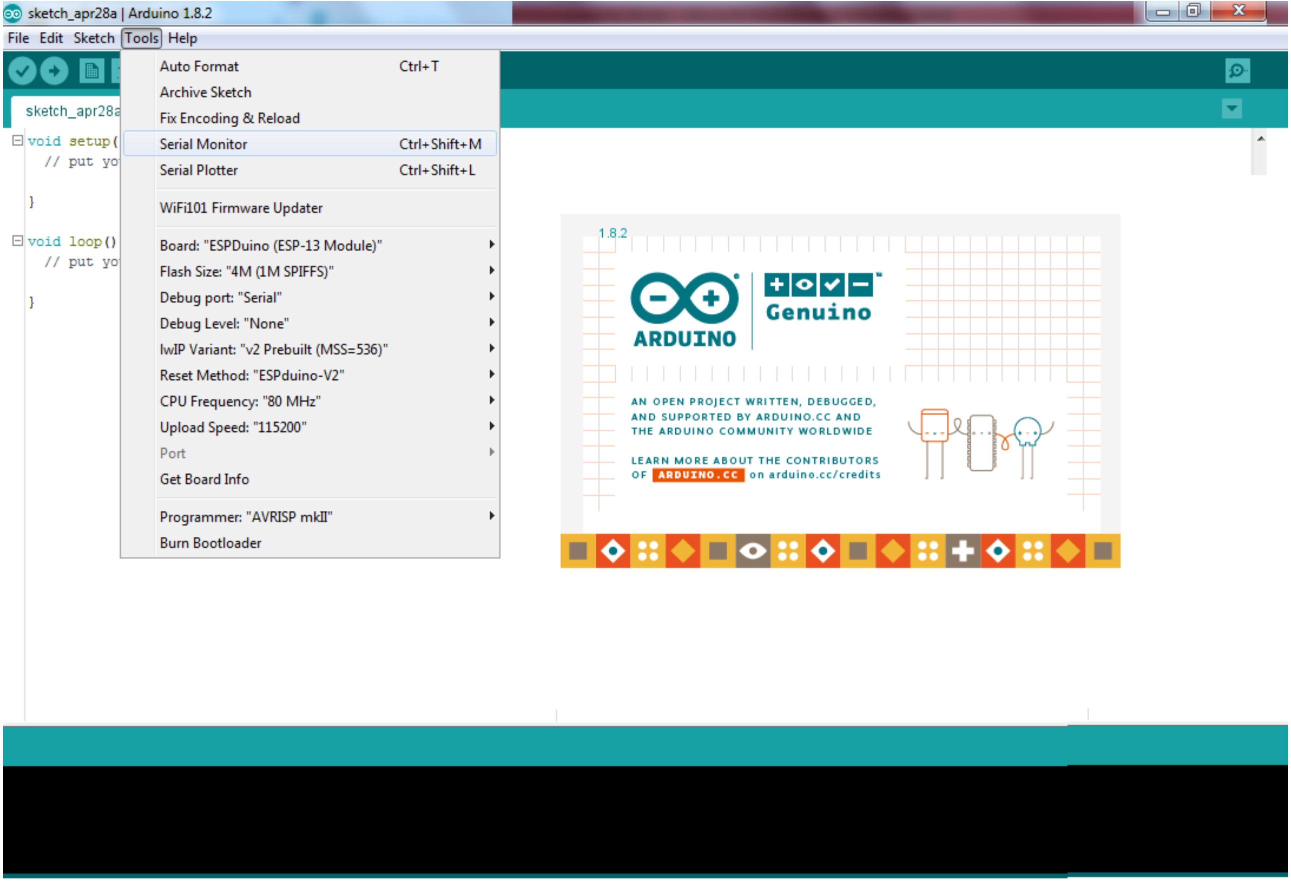
MAKE & COMPILE PROGRAM 22

CHAPTER-4: SOFTWARE PROGRAMMING OF ARDUINO

In this chapter the basic introduction Arduino software, IDE, basics of Programming and core program of project including its all program compilation steps.

4.1INTRODUCTION

The Arduino Integrated Development Environment is a cross-platform application that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards.


sketch_apr28a | Arduino 1.8.2
File
Edit
Sketch
Tools
Help
sketch_apr28a
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 Firmware Updater
Port
Get Board Info
Burn Bootloader
1.8.2
ARDUINO
Genuino
AN OPEN PROJECT WRITTEN, DEBUGGED, AND SUPPORTED BY ARDUINO.CC AND THE ARDUINO COMMUNITY WORLDWIDE
LEARN MORE ABOUT THE CONTRIBUTORS OF ARDUINO.CC on arduino.cc/credits

Board: "ESPDuino (ESP-13 Module)"
Flash Size: "4M (1M SPIFFS)"
Debug port: "Serial"
Debug Level: "None"
lwIP Variant: "v2 Prebuilt (MSS=536)"
Reset Method: "ESPduino-V2"
CPU Frequency: "80 MHz"
Upload Speed: "115200"
Programmer: "AVRISP mkII"

```
void setup(
  // put yo

}

void loop()
  // put yo

}
```

Figure (4.1): Startup screen of Arduino Software

After successful run of program Data can be sent to the Arduino from the Serial Monitor window in the Arduino IDE. A user can enter data in the input field in the serial monitor window to send values and data to the Arduino. To open serial monitor Ctrl+Shift+M can be used or alternatively we can go to tool bar as shown in figure.

It's Serial. begin(9600); passes the value 9600 to the speed parameter. This tells the Arduino to get ready to exchange messages with the Serial Monitor at a data rate of 9600 bits per second. That's 9600 binary ones or zeros per second, and is commonly called a baud rate. Serial.

If you try to do something like this: void loop() { digitalWrite(pin, LOW); // wrong: pin is not in scope here. } you'll get the same message as before: "error: 'pin' was not declared in this scope". That is, even though you've declared pin somewhere in your program, you're trying to use it somewhere outside its scope.

4.2PROGRAMMING LANUAGE

The Arduino IDE supports the languages C and C++ using special rules of code structuring.

Arduino programs are written in the Arduino Integrated Development Environment (IDE). Arduino IDE is a special software running on your system that allows you to write sketches (synonym for program in Arduino language) for different Arduino boards. The Arduino programming language is based on a very simple hardware programming language called processing, which is similar to the C language. After the sketch is written in the Arduino IDE, it should be uploaded on the Arduino board for execution.

The first step in programming the Arduino board is downloading and installing the Arduino IDE. The open source Arduino IDE runs on Windows, Mac OS X, and Linux. Download the Arduino software (depending on your OS) from the official website and follow the instructions to install.’

Now let’s discuss the basics of Arduino programming.

The structure of Arduino program is pretty simple. Arduino programs have a minimum of 2 blocks,

Preparation & Execution

Each block has a set of statements enclosed in curly braces:

```
void setup( )
{
statements-1;
.
.
.
statement-n;
}
void loop ( )
{
statement-1;
.
.
.
statement-n;
}
```

Here, setup ( ) is the preparation block and loop ( ) is an execution block.

The setup function is the first to execute when the program is executed, and this function is called only once. The setup function is used to initialize the pin modes and start serial communication. This function has to be included even if there are no statements to execute.

```
void setup ( )
{
pinMode (pin-number, OUTPUT); // set the ‘pin-number’ as output
pinMode (pin-number, INPUT); // set the ‘pin-number’ as output
}
```

After the setup ( ) function is executed, the execution block runs next. The execution block hosts statements like reading inputs, triggering outputs, checking conditions etc..

In the above example loop ( ) function is a part of execution block. As the name suggests, the loop( ) function executes the set of statements (enclosed in curly braces) repeatedly.

```
Void loop ( )
{
digitalWrite (pin-number,HIGH); // turns ON the component connected to ‘pin-number’
delay (1000); // wait for 1 sec
digitalWrite (pin-number, LOW); // turns OFF the component connected to ‘pin-number’
delay (1000); //wait for 1sec
}
```

Note: Arduino always measures the time duration in millisecond. Therefore, whenever you mention the delay, keep it in milli seconds.

Now, let’s take a giant leap and do some experiments with Arduino

Blinking the LED

Fade-in and fade-out the LED

In the process of experimenting with Arduino, writing the Arduino program is not the only important thing, building the breadboard circuit is equally important.

Let’s take a look at how the breadboard circuit has to be built for both the experiments.

Components required:

- ✓ Arduino UNO R3 -1
- ✓ Breadboard -1
- ✓ Breadboard connectors -3
- ✓ LED -1
- ✓ 1K resistor -1
- ✓ Blinking LED

Steps in building a breadboard connection:

Step-1: Connect the Arduino to the Windows / Mac / Linux system via a USB cable
Step-2: Connect the 13th digital pin of Arduino to the positive power rail of the breadboard and GND to the negative
Step-3: Connect the positive power rail to the terminal strip via a 1K ohm resistor
Step-4: Fix the LED to the ports below the resistor connection in the terminal strip
Step-5: Close the circuit by connecting the cathode (the short chord) of the LED to the negative power strip of the breadboard.

4.3ARDUINO CODING & IDE

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online.

The Arduino **Integrated Development Environment (IDE)** is the main text editing program used for Arduino programming. It is where you'll be typing up your code before uploading it to the board you want to program. Arduino code is referred to as sketches.

When we do a Serial print, the data you are trying to print is placed in an internal "transmit" buffer. Then, as each byte is transmitted by the hardware an interrupt is called (the "USART, Data Register Empty" interrupt) and the interrupt routine sends the next byte from the buffer out of the serial port.

To call a function, use the function name followed by opening and closing parentheses. Finally terminate the statement that calls the function with a semicolon. Load the sketch to an Arduino and then open the terminal window. The sketch prints some text in a box as shown below.

The green LED is marked with ON , and it indicates that the Arduino has power. The yellow LED is marked L and is just connected to pin 13. So when you set pin 13 to HIGH the LED lights up.

Arduino is neither a microcontroller nor a microprocessor. It is just a development board which contains a microcontroller mainly 8 bit AVR such as ATmega8,ATmega168,ATmega328,ATmega1280,ATmega2560.It is used to make so many college based projects easily.

In Short we can say that Arduino = Microcontroller + Crystal + on-board power supply pins, bootloader so that it can be programmed using a simple way through the Arduino IDE. In a nutshell, the main difference between the Arduino and a micro-controller is similar to the difference between a car and its engine.

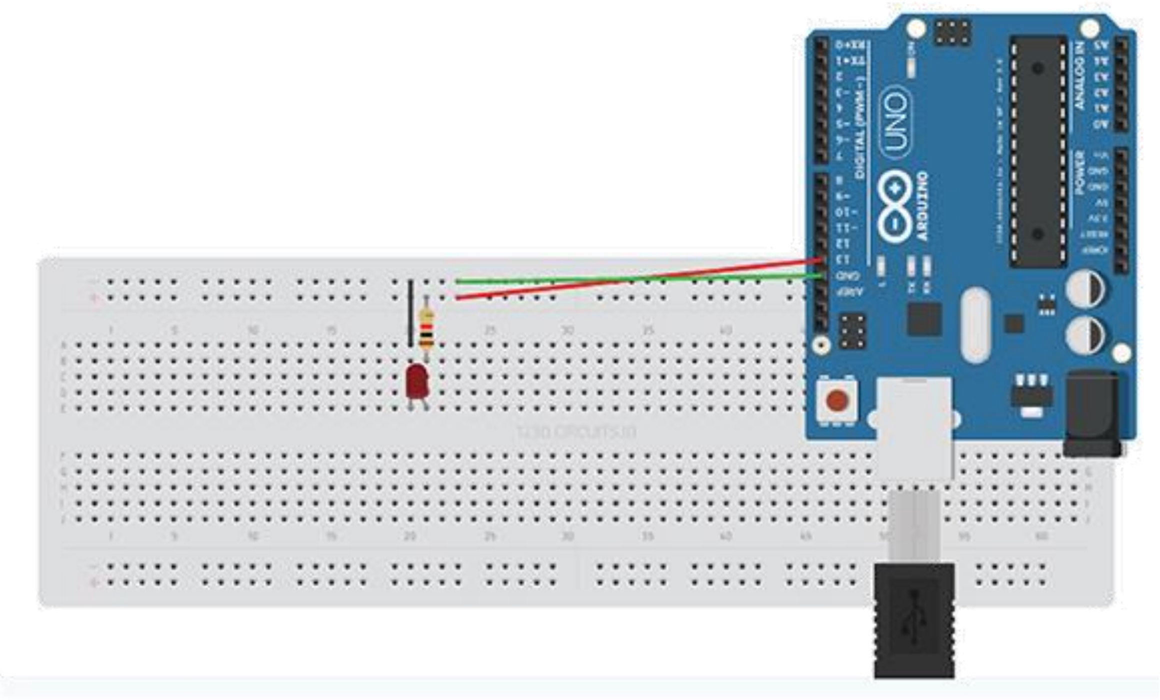
Figure: Signal of Program Compilation

4.4MAKE & COMPILE PROGRAM

The good news is that Arduino code is forgiving. It still has a learning curve, and it's not exactly easy, but it's definitely easier than trying to code your first mobile app or video game. Indeed, Arduino can be a stress-free way to wet your feet in coding. Of course, it's not perfect.

Getting Started with Arduino UNO

i. Use your Arduino Uno on the Arduino Web IDE.
ii. Use your Arduino Uno on the Arduino Desktop IDE. Install the board drivers. Open your first sketch. Select your board type and port. Upload the program. Learn more on the Desktop IDE.



iii. Tutorials.

The Logic of this code is fairly simple the values of potentiometers are stored in an array the records are then traversed using a for loop and the servos do the steps as per the values. You can check out this tutorial I used for reference "Arduino Potentiometer Servo Control & Memory"

The following steps are adopted to write a program for our project:

Step 1: First we will declare all the necessary variables globally so we can use them throughout the program. No special explanation is needed for this.

Step 2: Now we will write a setup function, where we set pins and their functions. This is the main function that executes first.

Step 3: Now we will write loop function:

:Programming:

```
/*

 * Record & Play robotic arm Code

 * byElectrical EnggPC

 * April 28, 2020

*/

#include <Servo.h>
```

```
Servo Servo_0;
Servo Servo_1;
Servo Servo_2;
Servo Servo_3;
.
int Pot_0;
int Pot_1;
int Pot_2;
int Pot_3;

int Servo_0_Pos;
int Servo_1_Pos;
int Servo_2_Pos;
int Servo_3_Pos;

int Prev_0_Pos;
int Prev_1_Pos;
int Prev_2_Pos;
int Prev_3_Pos;

int Current_0_Pos;
int Current_1_Pos;
int Current_2_Pos;
int Current_3_Pos;
int Storage[600];


int Index = 0;

char data = 0;


intServo_Position;

intServo_Number;


void setup()

{

Serial.begin(9600);


  Servo_0.attach(3);

  Servo_1.attach(5);

  Servo_2.attach(6);

  Servo_3.attach(9);


  Servo_0.write(100);

  Servo_1.write(100);

  Servo_2.write(100);

  Servo_3.write(100);


Serial.println("Press 'R' to Record and 'P' to play");

}
```

```
voidMap_Pot()

{

  Pot_0 = analogRead(A0);

  Servo_0_Pos = map(Pot_0, 0, 1023, 1, 179);

  Servo_0.write(Servo_0_Pos);



  Pot_1 = analogRead(A1);

  Servo_1_Pos = map(Pot_1, 0, 1023, 1, 179);

  Servo_1.write(Servo_1_Pos);



  Pot_2 = analogRead(A2);

  Servo_2_Pos = map(Pot_2, 0, 1023, 1, 179);

  Servo_2.write(Servo_2_Pos);



  Pot_3 = analogRead(A3);

  Servo_3_Pos = map(Pot_3, 0, 1023, 1, 179);

  Servo_3.write(Servo_3_Pos);



}



void loop()

{

Map_Pot();



while (Serial.available() > 0)
```

```
  {

data = Serial.read();

if (data == 'R')

Serial.println("Recording Moves...");

if (data == 'P')

Serial.println("Playing Recorded Moves...");

  }



if (data == 'R')

  {

    Prev_0_Pos = Servo_0_Pos;

    Prev_1_Pos = Servo_1_Pos;

    Prev_2_Pos = Servo_2_Pos;

    Prev_3_Pos = Servo_3_Pos;

    Prev_4_Pos = Servo_4_Pos;



Map_Pot();



if (abs(Prev_0_Pos == Servo_0_Pos))

    {

      Servo_0.write(Servo_0_Pos);



if (Current_0_Pos != Servo_0_Pos)

      {

Storage[Index] = Servo_0_Pos + 0;
```

```
        Index++;

      }



      Current_0_Pos = Servo_0_Pos;

    }



if (abs(Prev_1_Pos == Servo_1_Pos))

    {

      Servo_1.write(Servo_1_Pos);



if (Current_1_Pos != Servo_1_Pos)

      {

Storage[Index] = Servo_1_Pos + 100;

        Index++;

    }



      Current_1_Pos = Servo_1_Pos;

   }





if (abs(Prev_2_Pos == Servo_2_Pos))

   {

     Servo_2.write(Servo_2_Pos);



if (Current_2_Pos != Servo_2_Pos)
```

```
     {
Storage[Index] = Servo_2_Pos + 200;
       Index++;
     }

     Current_2_Pos = Servo_2_Pos;
   }



if (abs(Prev_3_Pos == Servo_3_Pos))
   {
     Servo_3.write(Servo_3_Pos);

if (Current_3_Pos != Servo_3_Pos)
     {
Storage[Index] = Servo_3_Pos + 300;
       Index++;
     }

     Current_3_Pos = Servo_3_Pos;
   }



Serial.print(Servo_0_Pos);
Serial.print(" \t ");
Serial.print(Servo_1_Pos);

Serial.print(" \t ");

Serial.print(Servo_2_Pos);

Serial.print(" \t ");

Serial.print(Servo_3_Pos);

Serial.print(" \t ");

Serial.println(Servo_4_Pos);

Serial.print ("Index = ");

Serial.println(Index);

delay(50);

  }



if (data == 'P')

    {

for (int i = 0; i < Index; i++)

  {

Servo_Number = Storage[i] / 100;

Servo_Position = Storage[i] % 100;



switch(Servo_Number)

  {

case 0:

        Servo_0.write(Servo_Position);

break;
case 1:

        Servo_1.write(Servo_Position);

break;



case 2:

        Servo_2.write(Servo_Position);

break;



case 3:

        Servo_3.write(Servo_Position);

break;



break;

    }

delay(50);

  }

 }

}
```

Once the code is ready, Now upload it to the arduino board.

After uploading the code to the board successfully, Open 'Serial Monitor' you can find it in Tools option. When Serial monitor starts the arduino will reset. Now you can control the robotic arm using the master arm. But nothing is being recorded.

To start recording, Enter 'R' in the monitor now you can perform the moves you wish to record.

After the moves are done you have to enter 'P' in order to play the recorded moves. The servos will continue to perform the moves as long as the board is not reset.

Chapter 5: CONCLUSION & FUTURE SCOPE **33-33**

CONCLUSION 33

FUTURE SCOPE 33

REFERENCE **34**

CHAPTER-5: CONCLUSION AND FUTURE SCOPE

In this chapter the basic introduction Arduino software, IDE, basics of Programming and core program of project including its all program compilation steps.

CONCOLUSION

Robotic arms, many areas are developable. Thanks to the robotic arms, many tasks are made easier and the resulting error level has been reduced to a minimum. For example; some pharmacy-based drug-giving robots, Medical Surgical robots and a projected robot arm have been developed. In addition to this, the ability to move the robot arm is further increased, and when the camera is placed in the finger area and the sensitivity is increased, it can be used in a wide range of applications from the medical sector to the automation systems. With the robotic arms developed in this way, the risk of infecting the patient in the medical sector is minimized, while the human errors are minimized during the surgical intervention. Despite the fact that the robotic arm made by this project is of prototype quality, it has a quality that can be improved for more robotic systems. Besides these, robotic arm sector, which is open to development, will keep its importance in the future.

The purpose of the project is to provide control of 3 axis moving robotic arm design and this robot arm with a suitable Arduino UNO and USB module through IDE software Simulator. The necessary theoretical and practical information for this purpose has been obtained and the necessary infrastructure has been established for the project. During the process of making and developing the project, a lot of theoretical knowledge has been transferred to the practice and it has been ensured that it is suitable for the purpose of the project.

FUTURE SCOPE

There's increased use of robotic hands in warehouses where they perform jobs from heavy lifting to inventory collection and other assorted jobs. The Indian market for warehouse automation is expected to grow at a CAGR of 10-12 percent between 2015-2020 and reach US $3.49bn by 2020. As the Robotics is concern that it is the world to learn & earn of the money, talent and innovation skills. According to the Bureau of Labor Statistics, demand for qualified robotics engineers is expected to grow by as much as 13 percent through 2018. To prepare workers, community colleges and universities are adding curriculum around robotic manufacturing aimed at undergrads and adult learners.

a) After a successful development of our prototype an Automatic crane can be developed to pic and place the bricks in sequential manner aiming to construct a wall or any construction of civil Engineering.

b) Small automated Robotic hands can be develop to handle micro particles such as tiny ICs, Tiny drug capsules. Human Brain veins etc.

c) Robotic surgery, or robot-assisted surgery, allows doctors to perform many types of complex procedures over tiny objects with more precision, flexibility and control than is only possible with smallest automated surgical robot. Robotic surgery is usually associated with minimally invasive surgery in which all procedures performed through tiny incisions.

REFERENCES

[1] W.M.H.W Kadir, RE Samin, BSK Ibrahim. Internet controlled a robotic arm. Procedia Engineering. 2012.

[2] MAK Yusoff, RE Samin, mobile robotic arm. Procedia Engineering. 2012.

[3] AM Al-Busaidi, Development of an educational environment for online control of a biped robot using MATLAB and Arduino, (MECHATRONICS), 9th France-Japan.2012.

[4] Servomotors and Industrial Control by R. Firoozian, 2nd Edition (2014) Published by Springer International Publishing

[5] R Krishna, GS Bala, SS ASC, BBP Sarma. Design and implementation of a robotic arm based on haptic technology. Int. J. of Eng. Research. 2012.

[6] Electric Electronic Technology-Step and Servo Motors, SVET, 2007.

[7] The Multiple-Function Intelligent Robotic Arms RongJyue Wang, Jun-Wei Zhang, Jia-Ming Xu, and Hsin-Yu Liu

[8] http://www.pathfindersdesignandtechnology.com/worksheets/build_a_robotic_arm.pdf

[9] Fundamentals of Robotic Mechanical Systems, 4th Edition(2014), by Jorge Angeles, Published by Springer

[10] Design and Implementation of a Wireless Gesture Controlled Robotic Arm with VisionLove Agarwal, Varnika Gaur, Puneet Verma

[11] Development of a robotic-arm controller by using hand gesture recognition Celik, I.B. ; Dept. of Electr. Electron. Eng., Dokuz Eylul Univ., Izmir, Turkey; Kuntalp, M.

[12] The development of six D.O.F. robot arm for intelligent robotJie-Tong Zou ; Dept. of Aeronaut. Eng., Nat. Formosa Univ., Huwei Township, Taiwan; Des-Hun Tu.

[13] https://create.arduino.cc/projecthub/WolfxPac/simple-and-smart-robotic-arm-using-arduino-1ceda6

[14] J. L. Raheja, R. Shyam, U. Kumar, P. B. Prasad, "Real-time robotic hand control using hand gestures", *ICMLC 2010 – 2nd Int. Conf. Mach. Learn. Comput.*, pp. 12-16, 2010.

[15] J. L. Raheja, R. Shyam, U. Kumar, P. B. Prasad, "Real-time robotic hand control using hand gestures", *ICMLC 2010 – 2nd Int. Conf. Mach. Learn. Comput.*, pp. 12-16, 2010.

BASE PAPER

International Journal of Computer Applications (0975 – 8887)
Volume 106 – No.10, November 2014

Dynamics of Robotic Arm

Abhishek Chavan
Dept. of Computer Engg., SITS, Pune

Abhishek Bhuskute
Dept. of Computer Engg., SITS, Pune

Anmol Jain
Dept. of Computer Engg., SITS, Pune

Neha Shinde
Dept. of Computer Engg., SITS, Pune

M. B. Salunke
Dept. of Computer Engg., SITS, Pune

ABSTRACT

One of the major areas of Research & Development (R&D) that has made a drastic improvement in Computer Science and electronics is „Automation‟ and „Artificial Intelligence‟. Autonomous Systems are self-governed and does not require any manual interventions. This paper presents an overview of previous developments and the working of Robotic arms along with its mathematical aspects. Arm assembly is used to supplement the robot‟s gestures and allows it to grasp and move objects. Arduino-board and other electronic circuits can be used to drive arm actuators by programming. A robotic arm can have a wide range of applications including pick-and-drop, imitating human arm, drawing objects with programmed intelligence, throwing light-weight objects based on mathematically governed functions. Once designed and implemented arm can be mounted to the mobile base so as to allow itself a complete plane of locomotion. The human arm is considered to have rotational ability mainly at shoulder, elbow and wrist. The actuators that control the shoulder and, to a lesser degree, the elbow have to carry the load of the entire arm, hand, and payload. The major simulative actions of robot can be implemented based on „Learn-and-Improve‟ baseline which is called as Reinforcement Learning. In nutshell, the robot can perform a task iteratively analyzing errors from its previous attempts and improving in every successive attempt. The algorithms involved in these actions decide complexity of operation.

Keywords

Robotic Arm, Reinforcement Learning, Autonomous.

1. INTRODUCTION

Currently there are many industrial applications which use the rigid, manually operated or programmed robotic systems. But the area of application still remains narrow. The task of handling manual robotic arm is considered tedious as the operations of all the motors should be synchronized. Hence automatic robotic motions are emphasized for industrial applications. That is the system is very less intuitive and this is the knack of the article, that is, to develop the system that would make the existing system more intuitive and user friendly. So the aim of the system design is to explore various possible areas of applications where the human-like robotic arm will prove helpful overcoming strength, speed and robustness limitations. The applications would typically involve imitation of human-arm motion, pick-and-drop action, throwing of object at specified target based on „Learn-and-Improve‟ method.

2. LITERATURE SURVEY-

Early and Modern Robots Robotic Arms:

Now, with the advent of machines, electronics, automation and the incorporation of solid-state transistors instead of vacuum tubes, the evolution of the microcircuit and more rapid computer systems, the stage was set for early modern robotic arm evolution. The first "position controlling apparatus" was patented in 1938 by Willard Pollard. This was a spray finishing robotic arm that had five degrees-of-freedom and an electrical control system. Although Pollard never built his arm, his design and interest in an mechanized application for automated robotic arms would spur on the ingenuity of others. Harold A. Roselund, working for De Vilbiss, developed another sprayer that was indeed manufactured. Both arms were useful and task oriented for their time and every movement at the respective joints in unique ways.

"The Multiple-Function Intelligent Robotic Arms"

The main design goal of this robotic arm was to present the following functions: Chinese calligraphy, fancy dancing, weightlifting and colour classification. Another design goal was to minimize cost and maximize performance.[1]

Hence it is designed in a way so as to perform many tasks as a human arm. The simplicity of its design gave it the way to perform motion as good as a human arm with improved values of strength and speed.[1]

"Development of a Robotic-Arm Controller by using Hand Gesture Recognition"

This robotic arm controller is using image processing in the field of Human-Machine Interaction. There are two methods used to control the robotic arm, the main task of them is to get the hand gesture data without using tool that helps the system to extract data easier (ex. glove or wrist band). The first method is comparing of all pre-stored data in the database at the Template Matching Algorithm, the second method is Signature Signal, distance signal between edge of the hand and centre of hand. These necessarily use the gyro-censors as the computation of change in angular values shall be obtained at time of operation.[5]

The subsequent operations are decided based on these changes.[5]

"The Development of Six D.O.F. Robot Arm for Intelligent Robot"

The aim of this paper is to develop a six D.O.F robot arm for an intelligent robot. As for the base or shoulder part, the harmonic responding drive, which has some advantages, was used to ensure the weight of the whole arm could be supported. This provides a degree of freedom along with supporting the assembly. The kinematic equations of the system have been studied and shown in this. The degrees in which the robotic unit make moves or acquires motion necessarily contribute to improvement of operation. Every added or updated motion represents ease of operation in certain operating context.[6]

At the end, we investigate some experiments; the computer system based controller can command the robot arm to pick a box through a moving path successfully.[6]

3. LITERATURE REVIEWED

Our system is an autonomous robotic arm which estimates trajectory and plans motion on it. We will be using a new concept called reinforcement learning which will enable our system to learn itself the path and trace it. For this we have reviewed some IEEE papers understanding and using them for our system. Some of the titles of these papers are listed below:

"Design and Implementation of Robotic Arm based on Haptic Technology"

This involves designing a haptic robotic arm, which can be used to pick and place the4 object. In this paper a robotic arm with four degrees of freedom is designed and is able to pick the objects with a specific weight and placed them in a desired location. To facilitate the lifting of the objects, Servomotors with a torque of 11kg are used. The programming is done on ATMEGA-328 Microcontroller using Arduino programming. The input is given using a remote, which is an arm, made of polycarbonate fitted with potentiometers with a certain angle of rotation. The potentiometers detect the angle of rotation and the signals are sent to the Microcontroller accordingly.

"Image processing and Recognition System for a Robot Arm Control"

Image processing (IP) and recognition system is based on continuous monitoring of environment with sensors and responding accordingly. These are considered intelligent as their behaviour is completely based on changes in environmental system and triggered associated programmed actions.

"The Development of Six D.O.F. Robot Arm for Intelligent Robot"

The motion is guided by the manually operated controller or a program that guides the appropriate motion and actions of robot. . The degrees in which the robotic unit make moves or acquires motion necessarily contribute to improvement of operation. Every added or updated motion represents ease of operation in certain operating context.[6]

Hence, the idea that a circular rotatable base may prove very useful as it is less costly and less time consuming for an assembly to rotate rather than to turn it. Care should be taken that this provision doesn‟t hamper any other DOF previously considered.[6]

4. METHEDOLOGIES

Throwing and Reinforcement Learning

When we, as humans, think of throwing something to some target, we don‟t really think much, we just throw. The fact is that, our brain is so used to such things that we don‟t need to think nor we do any calculation while throwing. But, with deeper observations we realized that, actually throwing is very complex and time critical task. It involves lots of calculations and factors and angular movements of the arm.

All these things, as a human, are very easy and we generally take this intelligence for granted. When it is to throw, we are not much concerned about the object we are holding, we just focus on the target. The brain approximates the force to apply, the angular motion of the hand and the timing to release the object, so that it will hit the target. It is a challenging task to implement this on a robotic arm. The system should be autonomous, taking the target information from the cameras and processing it to map all the necessary factors in place. Also defining and estimating the trajectory on which the object will travel is a critical task, as there is no way to control the object in air, once the object is released from the grip.

We suggest reinforcement learning for doing this task dynamically. The system will learn itself to throw and move towards precision to hit a target. Precise mapping of the data from camera to real world measure is most important. For this we suggest to start with some arbitrary factors and keep the record of the results of every attempt.

Initially, throwing with arbitrary values, the object will either go beyond the target or it may simply fall short. In case the object goes beyond the target set it as F_{max} (Maximum required value to hit the target). In some iteration, when the object drops before hitting target set it as F_{min} (Min required value to hit the target. Now, as you have F_{max} and F_{min}, use binary search to search a perfect scaling factor for mapping. It will take several iterations to get the perfect value. Once you get this perfect factor of scaling and mapping, change the target position, the arm will throw the object and hit the target at first shot. This process involves reinforcement learning in which system learns and improves itself. This makes the system dynamic and can work in various environments.

Application of this includes sorting of objects for industrial or domestic purpose; Also, playing games such as Dart. It will be real fun to play the dart against the robotic arm.

5. MATHEMATICAL MODEL

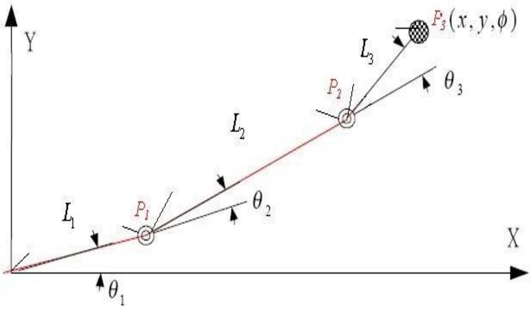


Figure 1

Considering 2-dimensional mathematical model so as to establish relational expression among θ1, θ2 and θ3 are the rotated angles of three links respectively. x and y are the final positions of the multi link system. Φ is summation angle of θ1, θ2 and θ3 displacement, angular displacement and obtained position, following relation can be considered.

L_1, L_2 and L_3 are the lengths of three links. In this case, the rotational matrices are as follows:

$$R_{ix} = \begin{matrix} 1 & 0 & 0 \\ 0 & \cos\theta_1 & -\sin\theta_1 \\ 0 & \sin\theta_1 & \cos\theta_1 \end{matrix}$$

$$R_{iy} = \begin{matrix} \cos\theta_1 & 0 & \sin\theta_1 \\ 0 & 1 & 0 \\ -\sin\theta_1 & 0 & \cos\theta_1 \end{matrix}$$

$$L_i = \begin{matrix} 0 \\ 0 \\ l_i \end{matrix}$$

$P_1=R_{1x}L_1$

$P_2=R_{1x}L_1 + R_{1X}R_{2x}L_2$

$P_3=R_{1x}L_1+ R_{1X}R_{2x}L_2 + R_{1x}R_{2x}R_{3x}L_3$

Where, P1, P2 and P3 are positions of the links.

6. FUTURE SCOPE

Robotic Arms has a wide scope of development. In the near future the arms will be able to perform every task as humans and in much better way. Imagination is the limit for its future applications.

It can be a real boon for handicapped people, who are paralyzed or lost their hands in some accident. The arm can be trained to listen to the command from a human and perform that task. A Precise gesture controlled system is also possible. Wearable devices can be used to send the command and control the movements of the arm.

Brain Computer Interface (BCI) is an immerging field of research. BCI can be used to acquire signals from the human brain and control the arm. The system can work in the same way as human arm. A person who may have lost his hand in any accident can resume his life like previous by such artificial arms. Robotic arms are versatile and have enormous ways of implementations.

7. CONCLUSION

This paper presents our study on early research and developments in robotic arms. It also explains basic mathematical model of the arms which can be used to implement the arm. Even after many years of research, the applications of robotic arm are restricted to the industries and primarily used in manufacturing units for increasing productivity. These arms are very sophisticated and can manage to make extremely precise movements. The robotic arms have wide variety of general purpose and domestic applications too, which are not much explored. Cost is the main constraint on robotic arms and to bring it down is a challenging issue. High torque servos with high precision are necessary for building these machines. These are the main components which cause the motion of the arm, and are most expensive. Finding alternatives to these motors to bring down the cost is the necessity. Also the material which will be used for the body should be light and durable. The light weight body can improve the performance of the motors and the torque. The shapes and size of the components and parts varies widely depending on the applications. Bringing these machines on product level for general purpose application is a tough job. If these constraints are resolved, soon the robotic arms will be available everywhere to serve as a “helping hand”.

8. REFERENCES

[1] The Multiple-Function Intelligent Robotic Arms Rong-Jyue Wang, Jun-Wei Zhang, Jia-Ming Xu, and Hsin-Yu Liu

[2] http://www.pathfindersdesignandtechnology.com/worksheets/build_a_robotic_arm.pdf

[3] A Low-cost Compliant 7-DOF Robotic ManipulatorMorgan Quigley, Alan Asbeck, and Andrew Y. Ng

[4] Design and Implementation of a Wireless Gesture Controlled Robotic Arm with VisionLove Agarwal, Varnika Gaur, Puneet Verma

[5] Development of a robotic-arm controller by using hand gesture recognition Celik, I.B. ; Dept. of Electr. Electron. Eng., Dokuz Eylul Univ., Izmir, Turkey; Kuntalp, M.

[6] The development of six D.O.F. robot arm for intelligent robotJie-Tong Zou ; Dept. of Aeronaut. Eng., Nat. Formosa Univ., Huwei Township, Taiwan; Des-Hun Tu.